

Pookie

I. Context

This project aims to create an assistant to play poker in online tournaments. Pookie by name, has to guide the player on the reaction he should have to maximise his chances to win. This way, the AI must suggest to fold, check, call, bet or raise depending on the situation, and indicate a chip amount in case of a bet or a rise.

▼ Here is an explanation of the basic rules of poker.

No Limit Texas Hold'em is a popular variation of poker played with a standard 52-card deck. The game can be played with anywhere from 2 to 10 players, and the objective is to win the pot, which is the sum of all bets made by the players in a hand.

At the beginning of the game, each player is dealt two private cards, known as hole cards. These cards are only visible to the player who receives them. Then, five community cards are dealt face-up in the middle of the table, and all players can use these cards to make the best possible five-card hand.

The game is played in rounds of betting, with the first round starting with the player to the left of the dealer, and continuing clockwise around the table. The first round of betting is called the pre-flop, and it begins with the player to the left of the big blind (which is a mandatory bet made by the player to the immediate left of the dealer).

During the pre-flop, each player has the option to fold (give up their hand), call (match the amount of the big blind), or raise (bet a higher amount). The betting continues until all players have either folded or called the highest bet.

After the pre-flop, the dealer deals the flop, which consists of three community cards dealt face-up in the middle of the table. Another round of betting takes place, starting with the player to the left of the dealer, and continuing clockwise.

Then, the dealer deals the turn, which is a fourth community card dealt face-up. Another round of betting takes place, starting with the player to the left of the dealer, and continuing clockwise.

Finally, the dealer deals the river, which is the fifth and final community card dealt face-up. Another round of betting takes place, starting with the player to the left of the dealer, and continuing clockwise.

If two or more players remain in the hand after the final round of betting, a showdown takes place, and the player with the best five-card hand using any combination of their hole cards and the community cards wins the pot. If there is a tie, the pot is split between the tied players.

In No Limit Texas Hold'em, players can bet any amount of chips they have in front of them at any time during the hand. This means that the amount a player can bet is only limited by the number of chips they have.

II. Database

1] Presentation

Noam Brown and Tuomas Sandholm recently published a [paper presenting Pluribus](#), the first no-limit non-heads-up poker AI that beat top professionals.

At the same time, they released the database delivering the course of a poker game for 10,000 hands. Here is a classic hand as it has been described in this dataset.

▼ PokerStars Hand #100000: Hold'em No Limit (50/100) - 2019/07/12 03:46:40 ET

Table 'Pluribus Session 100' 6-max (Play Money) Seat #6 is the button

Seat 1: MrBlue (10000 in chips)

Seat 2: MrBlonde (10000 in chips)

Seat 3: MrWhite (10000 in chips)

Seat 4: MrPink (10000 in chips)

Seat 5: MrBrown (10000 in chips)

Seat 6: Pluribus (10000 in chips)

MrBlue: posts small blind 50

MrBlonde: posts big blind 100

*** HOLE CARDS ***

Dealt to MrBlue [Tc Qc]

Dealt to MrBlonde [8s 4c]

Dealt to MrWhite [9c 3d]

Dealt to MrPink [Ah 4h]

Dealt to MrBrown [Th 5s]

Dealt to Pluribus [6c 7s]

MrWhite: folds

MrPink: raises 110 to 210

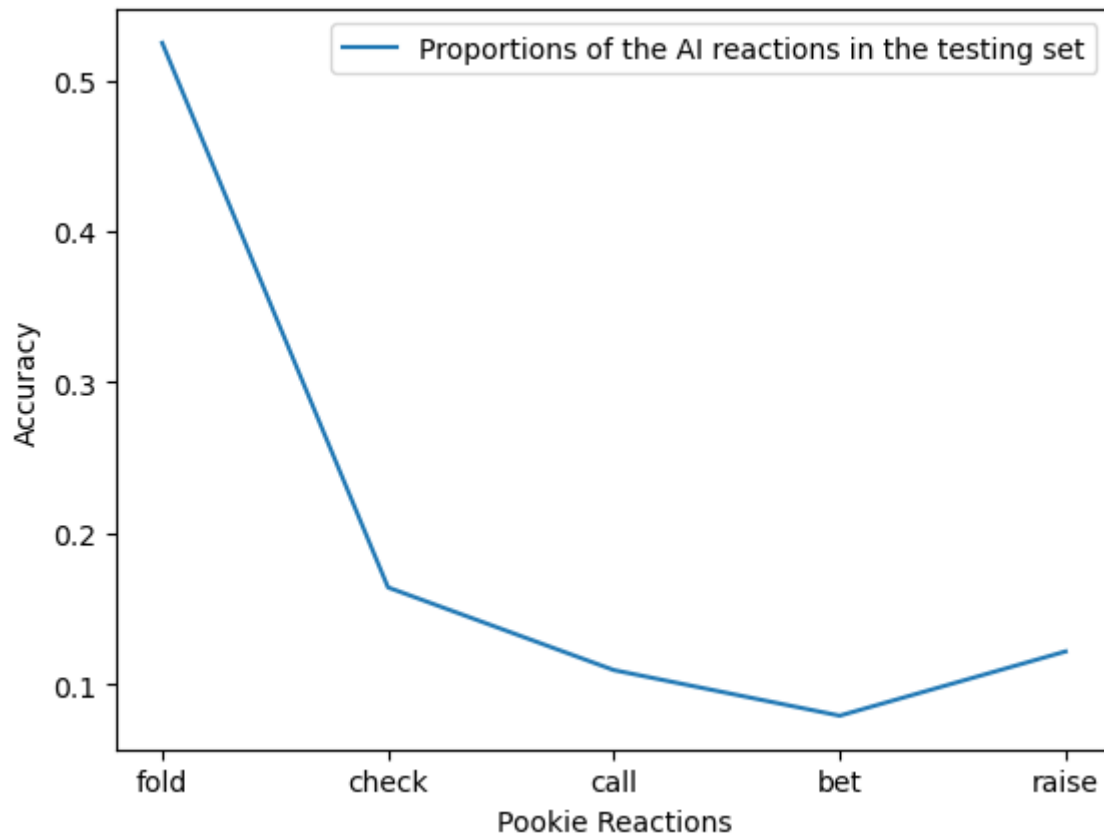
MrBrown: folds

Pluribus: folds

MrBlue: calls 160
MrBlonde: folds
*** FLOP *** [7d 5h 9d]
MrBlue: checks
MrPink: checks
*** TURN *** [7d 5h 9d] [7c]
MrBlue: checks
MrPink: checks
*** RIVER *** [7d 5h 9d] [7c] [Qh]
MrBlue: bets 230
MrPink: folds
Uncalled bet (230) returned to MrBlue
MrBlue collected 520.0 from pot
*** SUMMARY ***
Total pot 520 | Rake 0
Board [7d 5h 9d 7c Qh]

From this hand, we can extract the following data that will be directly used for the modelisation: the position of the player in the current hand, the small blind holder, the big blind older, the number of chips equivalent to the blinds, the cards in the hand of every player, the reaction of every player, the players' bets, the flop cards, the turn card and the river card.

2] Proportion of the issues in the testing set



```
Proportions of the AI reactions in the testing set :
fold : 0.5247660226588583
check : 0.16430408844617153
call : 0.10957254665863937
bet : 0.07941546713370916
raise : 0.12194187510262164
```

III. Features

1] Card

Pookie will have access to the nature of the cards it has in hands, and the ones shown on the table, that's to say the flop, the turn and the river in the hypothesis we've already reached this step of the current hand.

2] Players Reactions

We will consider in our modelisation all of the five possible reactions a player can have in a poker turn which is to fold, to check, to bet, to call and to rise. Moreover, Pookie ignores the reaction of the players who have to react after Pookie's turn, so

we will give them the status "NA". Finally, to distinguish the players still in the hand from the ones who fold during previous turns, we will attribute to these last ones the status "out".

3] Blind Amount

In a poker game, to avoid players refusing to play unless they have a monstrous hand, some players must pay a blind, which is the amount of chips that you have to bet even if you choose to fold before the flop. There are the big blind which is equal to the minimal bet and the small blind, which is equivalent to half the big blind. With each turn, both blinds are moving from a seat to the left. We will specify to Pookie who has to pay for the blinds and their values.

4] Position of the players

The first player to have his turn will be the one who is sitting at the left of the small blind. In a poker game, your position is an important criterion that you should take into account before making your decision. Indeed, if you're playing in the firsts, you ignore the reactions the other players will have, so you have to be more careful, and an example of strategy can consist in accepting only highly powerful hands. On the other side, if you're playing in the lasts, you can take more risks, a basic method consists in stealing the blinds, that's to say collecting the little amount of chips already on the table due to the blinds by making everyone else fold.

5] Stack, Chips on Table and Bet of the other players

In order to take a more accurate decision, Pookie will have access to the other player's stack, which are the chips every player has to his disposition in front of him. Of course, the objective is to collect the chips from everyone until their stack is empty. The "stack" feature is the number of chips the player has before his reaction. The chips on the table are the number of chips the player already has bet during this hand, before his reaction.

Finally, the bet is the number of chips the player will bet in accordance with his reaction, 0 for a fold or a check and the exact amount if the player decided to call, bet or raise.

6] Victory probability

One more feature we decided to add because it improved Pookie's results is the probability of victory determined by another artificial intelligence, that will take into

account the cards you have in hand and on the table if there are some. Based on this, it will tell you the probability you have to win. We could have done it by a recursive algorithm by testing each possibility to have a perfectly accurate victory probability but it was way too time-consuming.

7] Minimum Bet

mimum bet pour pluribus

puis minimum bet pour chaque joueur

IV. Model

1] Multi Layer Perceptron

I choose the "Multi-layer perceptron". This is the most widely studied and used neural network classifier. It can model complex functions, it is robust (it can ignore irrelevant inputs and noise) and it can adapt its weights and/or topology in response to changes in the environment. Another reason why we will use this type of perceptron is that it implements the black box view and can be used with little knowledge of the nature of the function being modelled.

I decided to model the network architecture with a single layer. As for the number of units in the layer, I studied the evolution of the accuracy of the model as a function of the number of neurons to determine this number. Networks with many hidden neurons can represent functions of any shape, but this flexibility can cause the network to learn noise from the data. This is known as "overtraining".

2] Reaction Model (RM)

Features

The RM is taking into account the player's position around the table, the amount of the small blind, the amount of the big blind, all available cards, in your hands, the flop cards, the turn card and the river card as well as the victory probability in favour of the player resulting from these cards and finally, all the other player's reactions before the concerned player to predict the reaction the player should have considering the situation.

Activation function

We choose the **Softmax function** which takes as input a vector of n real numbers and normalizes it to a probability distribution of n probabilities.

Cost function

We choose the **categorical_crossentropy** cost function to compare the estimated and actual probability distributions on Y . Used as a loss function for multi-class classification model where there are two or more output labels. The output label is assigned one-hot category encoding value in form of 0s and 1. The output label, if present in integer form, is converted into categorical encoding using keras.

Optimization function

We choose Adam's **optimization algorithm** which is an extension of the stochastic gradient descent.

3] Bet Model (BM)

Features

The BM is taking into consideration the player's position around the table, the amount of the small blind, the amount of the big blind, all available cards, in your hands, the flop cards, the turn card and the river card as well as the victory probability in favour of the player resulting from these cards and all the other player's reactions before the concerned player. That far, there are the same as for the RM, we will add the stack of every player, the chips they already put on the table before their reaction and the amount of their bet to predict the number of chips to bet considering the situation if the player reacted by betting or by raising.

Cost function

Mean squared error (MSE) loss is a widely-used loss function in machine learning and statistics that measures the average squared difference between the predicted values and the actual target values. It is particularly useful for regression problems, where the goal is to predict continuous numerical values.

$$MSE = \frac{1}{n} \sum \left(\underbrace{y - \hat{y}}_{\substack{\text{The square of the difference} \\ \text{between actual and} \\ \text{predicted}}} \right)^2$$

Optimization function

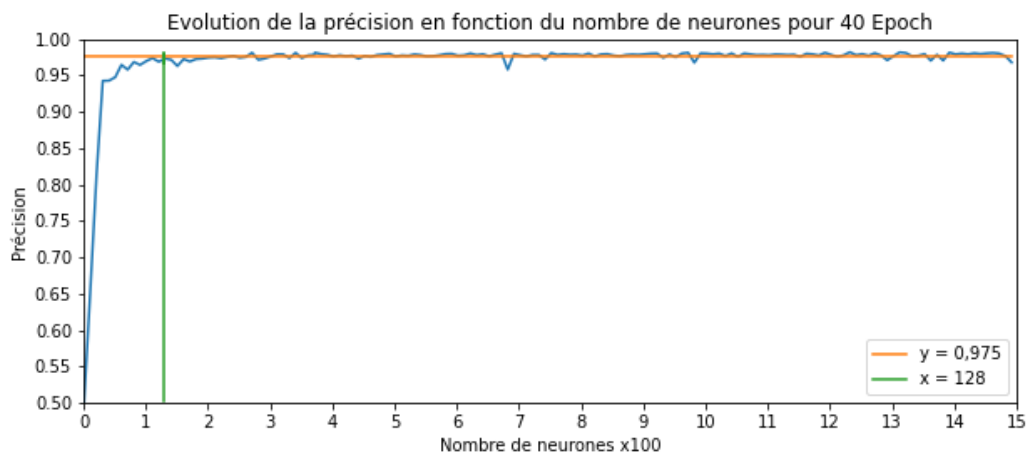
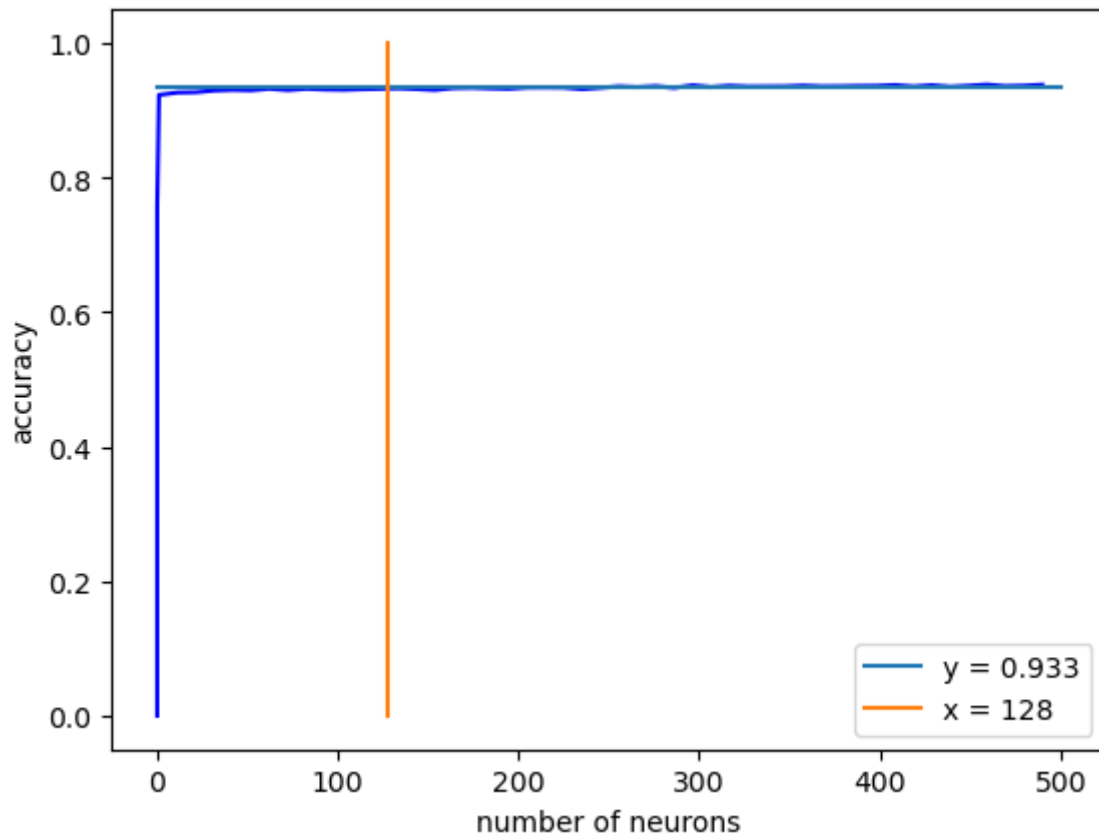
We choose Adam's **optimization algorithm** which is an extension of the stochastic gradient descent.

V. Results

1] Reaction Model

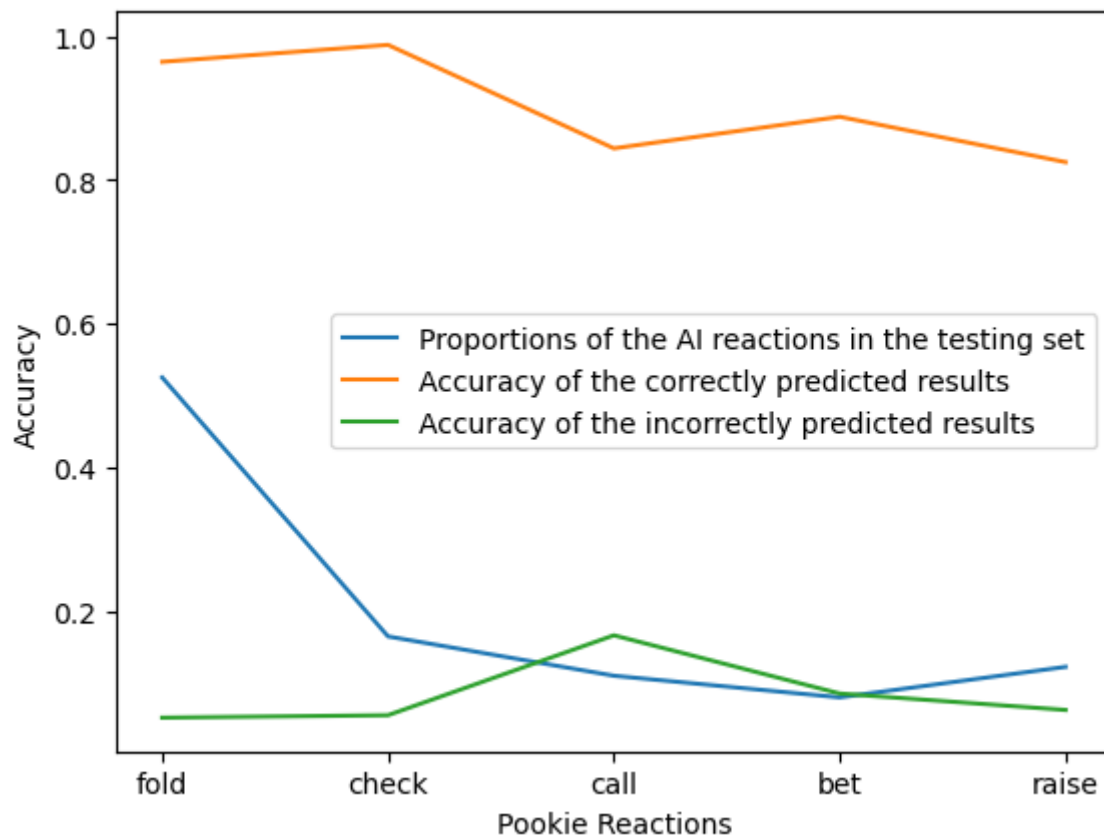
The RM has been trained and tested with data from 14 different high-ranked players, and each of them has a different strategy. There is a slight difference in their reactions but we should be able to draw a trend. This mix enables Pookie to produce better results even if it becomes harder to measure them.

Evolution of the accuracy depending on the number of neurons



By calculating the evolution of the precision according to the number of neurons, we note that after a hundred neurons, the precision converges towards 93.4%. We therefore chose to implement a network of 128 neurons.

Performances presentation



The orange curve represents the accuracy of Pookie based on the results she managed to identify and the green one represents the correctness of Pookie's predictions. To illustrate the difference let's imagine that Pookie is always suggesting to fold no matter the conditions, then she will manage to identify all the fold decisions so her accuracy would be high but on the other hand, she's always giving the same answer so her correctness would be low, because a large proportion of her "fold" predictions would be incorrect.

As we can see, she can easily detect when you are supposed to fold with 96% of the fold results identified and 95% of the fold predictions verified.

Pookie's performances show that she's competent when it comes to check with 99% of the check results identified and 95% of the check predictions verified.

On the other hand, it will be more difficult for Pookie to make sure when you have to call with 84% of the call results identified and 83% of the call predictions verified.

However, it looks like she's better at advising in a bet situation with 89% of the bet results identified and 92% of the bet predictions verified.

Ultimately, you should be careful about raising because it appears she can miss some opportunities to raise with only 83% of the raise results identified and 94% of the raise predictions verified.

Conclusion

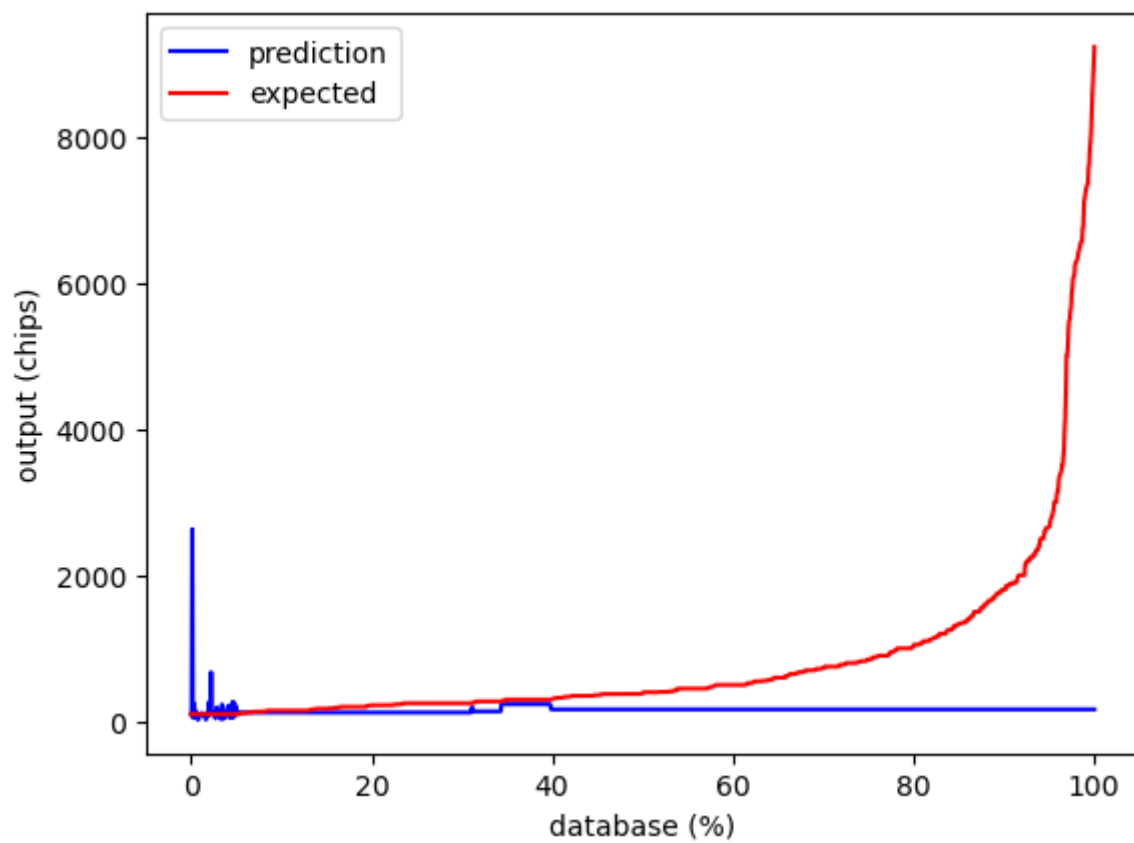
There are three main points these performances are showing.

1. Pookie's highly reliable when it comes to low-risk situations or low-risk predictions. However, you have to be more careful in high-risk situations or with high-risk predictions.
2. Pookie's ability to detect call situations and her capacity to deliver correct call predictions are below her other performances. Then, we will give it a deeper analysis. It appears that when she's confronted with a call situation she can't identify, Pookie will deliver a fold prediction 90% of the time and a raise prediction only 10% of the time. Also, when she's wrongly suggesting to call, nine out of ten, it was a fold situation and one out of ten, it was a raise situation. Then, it appears that these performances don't have a significant impact on the game.
3. Usually, the accuracy and the correctness are quite similar except for the "raise" reaction. We would better investigate. In front of a raise situation, Pookie will give 54% of fold instructions, 12% of check instructions, 8% of call instructions and 26% of bet instructions. We can perfectly explain this. There is only one situation when a player would raise with a very low victory probably and that's the kind of situation Pookie would hardly predict: the bluff. Therefore, it looks like the incorrect predictions won't have terrible consequences with half of these instructions as fold. The only thing you might lose from time to time is the opportunity to bluff. But this low performance comes from the fact that different high-ranked players' strategies have been used to train and test Pookie, and the bluff is a very personal strategy.

2] Bet Model

a) Bet

Final score (RMSE): 626.3499705524524



b) Raise

Final score (RMSE): 404.1417883065659

